

# Package ‘normfluodbf’

November 28, 2023

**Title** Cleans and Normalizes FLUOstar DBF and DAT Files

**Version** 1.5.2

**Description** Cleans and Normalizes FLUOstar DBF and DAT Files obtained from liposome flux assays. Users should verify extended usage of the package on files from other assay types.

**License** MIT + file LICENSE

**Encoding** UTF-8

**RoxygenNote** 7.2.3

**Suggests** knitr, learnr, rmarkdown, testthat (>= 3.0.0)

**Config/testthat/edition** 3

**Imports** data.table, foreign, tidyr, tibble, dplyr, emojiFont, rio, stats, ggplot2, ggthemes, badger, stringr

**Depends** R (>= 2.10)

**LazyData** true

**URL** <https://github.com/AlphaPrime7/normfluodbf>,  
<https://alphaprime7.github.io/normfluodbf/>

**BugReports** <https://github.com/AlphaPrime7/normfluodbf/issues>

**VignetteBuilder** knitr

**NeedsCompilation** no

**Author** Tingwei Adeck [aut, cre],  
normfluodbf authors [cph] (<<https://orcid.org/0009-0009-7450-8863>>)

**Maintainer** Tingwei Adeck <[awesome.tingwei@outlook.com](mailto:awesome.tingwei@outlook.com)>

**Repository** CRAN

**Date/Publication** 2023-11-28 22:10:03 UTC

## R topics documented:

actual_cols_used . . . . .	3
check_dat . . . . .	3
check_dbf . . . . .	4

clean_odd	5
clean_odd_cc	6
comma_cleaner	6
dat_1	7
dat_2	8
dat_3	8
dat_4	8
dat_5	9
dat_6	9
dat_7	9
dat_col_names_horizontal	10
dat_col_names_optimus	11
dat_col_names_prime	12
dat_col_names_rigid	13
decimal_scaling	15
fluor_threshold_check	16
fluor_threshold_check_na	17
fluor_threshold_check_raw	18
generic_identifier	19
ggplot_tnp	19
liposomes_214	20
liposomes_215	21
liposomes_216	21
liposomes_218	21
liposomes_221	22
liposomes_227	22
log_transformation	22
min_max_norm	23
min_max_norm_percent	24
normfluodat	25
normfluodatfull	26
normfluodatlite	28
normfluodat	29
normfluordbf	31
norm_applier	32
norm_tidy_dbf	33
norm_z	34
resample_dat	35
resample_dat_alt	36
resample_dat_scale	37
resample_dat_scale_alt	38
resample_dat_scale_alt_bfv	39
resample_dat_scale_alt_bf_na	40
resample_dat_scale_alt_na	41
resample_dat_scale_naretainer	42
resample_dat_scale_optimus	43
resample_dat_scale_optimus_backend	44
resample_dat_scale_optimus_na	45

<i>actual_cols_used</i>	3
roundfluor . . . . .	46
time_attribute . . . . .	46
unique_identifier . . . . .	47
<b>Index</b>	<b>49</b>

---

<code>actual_cols_used</code>	<i>Title: A function to get the actual columns used in the assay.</i>
-------------------------------	---

---

**Description**

A function that facilitates a users' workflow by helping extract the actual columns used in the assay.

**Usage**

```
actual_cols_used(dat)
```

**Arguments**

<code>dat</code>	A string ("dat_1.dat") if the file is found within the present working directory (pwd) OR a path pointing directly to a ".dat" file.
------------------	--

**Value**

Returns a numeric vector denoting the columns used in the assay.

**Author(s)**

Tingwei Adeck

**Examples**

```
fpath <- system.file("extdata", "dat_1.dat", package = "normfluodbf", mustWork = TRUE)
acutest <- actual_cols_used(fpath)
```

---

<code>check_dat</code>	<i>Title: A function to check for DATs in a directory.</i>
------------------------	--

---

**Description**

A function that facilitates a users' workflow by helping to check for DATs in a directory.

**Usage**

```
check_dat(pathstring)
```

**Arguments**

pathstring      A string for a path to a directory containing files.

**Value**

Returns a list of DAT files or a warning.

**Author(s)**

Tingwei Adeck

**Examples**

```
## Not run:  
fpath <- system.file("extdata", package = "normfluodbf", mustWork = TRUE)  
setwd(fpath)  
check_dat(getwd())  
  
## End(Not run)
```

---

check\_dbf

*Title: A function to check for DBFs in a directory.*

---

**Description**

A function that facilitates a users' workflow by helping to check for DBFs in a directory.

**Usage**

```
check_dbf(pathstring)
```

**Arguments**

pathstring      A string for a path to a directory containing files.

**Value**

Returns a list of DBF files or a warning.

**Author(s)**

Tingwei Adeck

## Examples

```
## Not run:
fpath <- system.file("extdata", package = "normfluodbf", mustWork = TRUE)
setwd(fpath)
check_dbf(getwd())

## End(Not run)
```

---

clean\_odddat\_optimus *Title: DAT file data frame cleaner.*

---

## Description

The function takes the dirty data frame obtained from reading the FLUOstar DAT file, applies an original algorithm that inserts NAs in place of the special characters, and then applies a function called `comma_cleaner()` to the dirty data frame for the removal of commas, and finally, rows with NAs only are removed.

## Usage

```
clean_odddat_optimus(df)
```

## Arguments

`df` A dirty data frame obtained from the FLUOstar DAT file.

## Value

A clean data frame with clean NA values retained.

## Author(s)

Tingwei Adeck

## See Also

[comma\\_cleaner\(\)](#), [clean\\_odd\\_cc\(\)](#)

## Examples

```
fpath <- system.file("extdata", "dat_1.dat", package = "normfluodbf", mustWork = TRUE)
dat_df <- read.table(file=fpath)
partial_cleaned_dat <- clean_odddat_optimus(dat_df)
```

---

clean\_odd\_cc                      *Title: DAT file data frame cleaner.*

---

**Description**

The function takes the dirty data frame obtained from reading the FLUOstar DAT file and applies a function called `comma_cleaner()` to the dirty data frame, which automatically inserts NAs in place of the special characters, and rows with NAs only are removed.

**Usage**

```
clean_odd_cc(df)
```

**Arguments**

`df`                      A dirty data frame obtained from the FLUOstar DAT file.

**Value**

A clean data frame with clean NA values retained.

**Author(s)**

Tingwei Adeck

**See Also**

[comma\\_cleaner\(\)](#), [clean\\_odddat\\_optimus\(\)](#)

**Examples**

```
fpath <- system.file("extdata", "dat_3.dat", package = "normfluodbf", mustWork = TRUE)
dat_df <- read.table(file=fpath)
cleaned_dat <- clean_odd_cc(dat_df)
```

---

comma\_cleaner                      *Title: Comma Cleaner function.*

---

**Description**

This modular function, in the context of this package, is responsible for removing commas from attribute(s) values. Removal of commas facilitates the conversion of attributes into the numeric class.

**Usage**

```
comma_cleaner(comma_df)
```

**Arguments**

comma\_df            A dirty data frame obtained from the FLUOstar DAT file.

**Value**

A clean data frame with numeric no-comma values for attribute(s).

**Author(s)**

Tingwei Adeck

**See Also**

[clean\\_odd\\_cc\(\)](#), [clean\\_odddat\\_optimus\(\)](#)

**Examples**

```
fpath <- system.file("extdata", "dat_1.dat", package = "normfluodbf", mustWork = TRUE)
dat_df <- read.table(file=fpath)
nocomma_dat <- comma_cleaner(dat_df)
```

---

dat_1	<i>dat_1.</i>
-------	---------------

---

**Description**

FLUOstar .dat files used for creation of the update and unusable for immediate data analysis.

**Usage**

```
dat_1
```

**Format**

An object of class `data.frame` with 320 rows and 12 columns.

---

dat_2	<i>dat_2.</i>
-------	---------------

---

**Description**

FLUOstar .dat files used for creation of the update and unusable for immediate data analysis.

**Usage**

dat\_2

**Format**

An object of class `data.frame` with 320 rows and 12 columns.

---

dat_3	<i>dat_3.</i>
-------	---------------

---

**Description**

FLUOstar .dat files used for creation of the update and unusable for immediate data analysis. This file is unique because it validates a major bug fix to ensure that users get the right output.

**Usage**

dat\_3

**Format**

An object of class `data.frame` with 320 rows and 12 columns.

---

dat_4	<i>dat_4.</i>
-------	---------------

---

**Description**

FLUOstar .dat files used for creation of the update and unusable for immediate data analysis. This file is unique because it validates a major bug fix to ensure that users get the right output.

**Usage**

dat\_4

**Format**

An object of class `data.frame` with 320 rows and 1 columns.



---

dat_5	<i>dat_5.</i>
-------	---------------

---

**Description**

FLUOstar .dat files used for creation of the update and unusable for immediate data analysis. This file is unique because it validates a major bug fix to ensure that users get the right output.

**Usage**

dat\_5

**Format**

An object of class `data.frame` with 105 rows and 1 columns.

---

dat_6	<i>dat_6.</i>
-------	---------------

---

**Description**

FLUOstar .dat files used for creation of the update and unusable for immediate data analysis. This file is unique because it validates a major bug fix to ensure that users get the right output.

**Usage**

dat\_6

**Format**

An object of class `data.frame` with 105 rows and 2 columns.

---

dat_7	<i>dat_7.</i>
-------	---------------

---

**Description**

FLUOstar .dat files used for creation of the update and unusable for immediate data analysis. This file is unique because it validates a major bug fix to ensure that users get the right output.

**Usage**

dat\_7

**Format**

An object of class `data.frame` with 105 rows and 3 columns.

---

`dat_col_names_horizontal`*Title: Attribute(s) naming function.*

---

### Description

This function is used to name attribute(s). Attribute(s) names, in this case, are equivalent to the well labels found on the microplate reader. An attribute for a sample loaded into row A - column 1 will be named A1. In short, the function takes a clean data frame and returns attribute names that match the FLUOstar plate layout often presented as an Excel file.

### Usage

```
dat_col_names_horizontal(dat = NULL, df, rows_used = NULL, cols_used = NULL)
```

### Arguments

<code>dat</code>	A string ("dat_1.dat") if the file is found within the present working directory (pwd) OR a path pointing directly to a ".dat" file.
<code>df</code>	A data frame that requires attribute labels.
<code>rows_used</code>	A character vector indicating the rows or tuples used on the microplate (usually a 96-well microplate). Initialized as NULL.
<code>cols_used</code>	A numeric vector indicating the plate columns or attributes used. Initialized as NULL.

### Value

Returns a character or numeric vector of attribute(s) names for the normalized data frame.

### Note

This function was designed to avoid the use of stringr. This function is designed to name attributes when the read direction is specified as horizontal.

### Author(s)

Tingwei Adeck

### Examples

```
fpath <- system.file("extdata", "dat_1.dat", package = "normfluodbf", mustWork = TRUE)
dat_df <- read.table(file=fpath)
nocomma_dat <- clean_odddat_optimus(dat_df)
resampled_scaled <- resample_dat_scale(nocomma_dat, tnp=3, cycles=40)
n = c('A', 'B', 'C')
sample_col_names <- dat_col_names_horizontal(dat=fpath, resampled_scaled, n)
```

---

dat\_col\_names\_optimus *Title: Attribute(s) naming function.*

---

### Description

This function is used to name attribute(s). Attribute(s) names, in this case, are equivalent to the well labels found on the microplate reader. An attribute for a sample loaded into row A - column 1 will be named A1. In short, the function takes a clean data frame and returns attribute names that match the FLUOstar plate layout often presented as an Excel file.

### Usage

```
dat_col_names_optimus(  
  dat = NULL,  
  df,  
  rows_used = NULL,  
  cols_used = NULL,  
  user_specific_labels = NULL,  
  read_direction = NULL  
)
```

### Arguments

dat	A string ("dat_1.dat") if the file is found within the present working directory (pwd) OR a path pointing directly to a ".dat" file.
df	A data frame that requires attribute labels.
rows_used	A character vector indicating the rows or tuples used on the microplate (usually a 96-well microplate). Initialized as NULL.
cols_used	A numeric vector indicating the plate columns or attributes used. Initialized as NULL.
user_specific_labels	A character vector where the user manually enters the used microplate wells based on the FLUOstar plate layout.
read_direction	A string input with two choices, "vertical" or "horizontal." The user indicates "vertical" if the user intends to have a final data frame with samples arranged as sample type triplets (A1, B1, C1, A1, B1, C1) OR "horizontal" if the user intends to have a final data frame with samples

### Value

Returns a character or numeric vector of attribute(s) names for the normalized data frame.

**Note**

Users are advised to input rows used but won't be penalized for not doing so. If the user provides the rows used, then attribute names are generated for the user. The user must check to ensure that the names match the microplate layout.

The user can leave the columns used as NULL if the user loaded samples from column 1 and did so in sequence. If the user fails to load in sequence from the first position, then the user must provide a numeric vector of columns used.

For instance, where the user skips columns, the user will be prompted to interact with the program in order to ensure the final data frame has the correct attribute names.

The user can bypass the rows used and columns used parameters if the user supplies a manually created character vector of the wells used in an experiment.

The read direction parameter is used to determine the presentation of the samples in the final data frame.

**Author(s)**

Tingwei Adeck

**See Also**

[normfluodat\(\)](#), [dat\\_col\\_names\\_rigid\(\)](#)

**Examples**

```
fpath <- system.file("extdata", "dat_1.dat", package = "normfluodbf", mustWork = TRUE)
dat_df <- read.table(file=fpath)
nocomma_dat <- clean_odddat_optimus(dat_df)
resampled_scaled <- resample_dat_scale(nocomma_dat, tnp=3, cycles=40)
n = c('A', 'B', 'C')
sample_col_names <- dat_col_names_optimus(dat = fpath, resampled_scaled, n)
```

---

dat\_col\_names\_prime     *Title: Attribute(s) naming function.*

---

**Description**

This function is used to name attribute(s). Attribute(s) names, in this case, are equivalent to the well labels found on the microplate reader. An attribute for a sample loaded into row A - column 1 will be named A1. In short, the function takes a clean data frame and returns attribute names that match the FLUOstar plate layout often presented as an Excel file.

**Usage**

```
dat_col_names_prime(
  dat = NULL,
  df,
  rows_used = NULL,
  cols_used = NULL,
  user_specific_labels = NULL
)
```

**Arguments**

dat	A string ("dat_1.dat") if the file is found within the present working directory (pwd) OR a path pointing directly to a ".dat" file.
df	A data frame that requires attribute labels.
rows_used	A character vector indicating the rows or tuples used on the microplate (usually a 96-well microplate). Initialized as NULL.
cols_used	A numeric vector indicating the plate columns or attributes used. Initialized as NULL.
user_specific_labels	A character vector where the user manually enters the used microplate wells based on the FLUOstar plate layout.

**Value**

Returns a character vector of attribute(s) names for the normalized data frame.

**Author(s)**

Tingwei Adeck

**Examples**

```
fpath <- system.file("extdata", "dat_1.dat", package = "normfluodbf", mustWork = TRUE)
dat_df <- read.table(file=fpath)
nocomma_dat <- clean_odddat_optimus(dat_df)
resampled_scaled <- resample_dat_scale(nocomma_dat, tnp=3, cycles=40)
n = c('A', 'B', 'C')
sample_col_names <- dat_col_names_prime(dat = fpath, resampled_scaled, n)
```

---

dat\_col\_names\_rigid     *Title: Attribute(s) naming function.*

---

**Description**

This function is used to name attribute(s). Attribute(s) names, in this case, are equivalent to the well labels found on the microplate reader. An attribute for a sample loaded into row A - column 1 will be named A1. In short, the function takes a clean data frame and returns attribute names that match the FLUOstar plate layout often presented as an Excel file.

**Usage**

```
dat_col_names_rigid(
  dat = NULL,
  df,
  rows_used = NULL,
  cols_used = NULL,
  user_specific_labels = NULL,
  read_direction = NULL
)
```

**Arguments**

<code>dat</code>	A string ("dat_1.dat") if the file is found within the present working directory (pwd) OR a path pointing directly to a ".dat" file.
<code>df</code>	A data frame that requires attribute labels.
<code>rows_used</code>	A character vector indicating the rows or tuples used on the microplate (usually a 96-well microplate). Initialized as NULL.
<code>cols_used</code>	A numeric vector indicating the plate columns or attributes used. Initialized as NULL.
<code>user_specific_labels</code>	A character vector where the user manually enters the used microplate wells based on the FLUOstar plate layout.
<code>read_direction</code>	A string input with two choices, "vertical" or "horizontal." The user indicates "vertical" if the user intends to have a final data frame with samples arranged as sample type triplets (A1, B1, C1, A1, B1, C1) OR "horizontal" if the user intends to have a final data frame with samples arranged as clusters per sample type (A1, A2, A3, B1, B2, B3).

**Value**

Returns a character vector of attribute(s) names for the normalized data frame.

**Note**

Users are advised to input rows used but won't be penalized for not doing so. If the user provides the rows used, then attribute names are generated for the user. The user must check to ensure that the names match the microplate layout.

The user can leave the columns used as NULL if the user loaded samples from column 1 and did so in sequence. If the user fails to load in sequence from the first position, then the user must provide a numeric vector of columns used.

For instance, where the user skips columns, the user will be prompted to interact with the program in order to ensure the final data frame has the correct attribute names.

The user can bypass the rows used and columns used parameters if the user supplies a manually created character vector of the wells used in an experiment.

The read direction parameter is used to determine the presentation of the samples in the final data frame.

This naming function only returns a character vector hence the rigid suffix.

**Author(s)**

Tingwei Adeck

**See Also**

[dat\\_col\\_names\\_optimus\(\)](#)

**Examples**

```
fpath <- system.file("extdata", "dat_1.dat", package = "normfluodbf", mustWork = TRUE)
dat_df <- read.table(file=fpath)
nocomma_dat <- clean_odddat_optimus(dat_df)
resampled_scaled <- resample_dat_scale(nocomma_dat, tnp=3, cycles=40)
n = c('A', 'B', 'C')
sample_col_names <- dat_col_names_rigid(dat = fpath, resampled_scaled, n)
```

---

decimal\_scaling

*Title: A decimal scaling function (a machine learning tool).*

---

**Description**

Title: A decimal scaling function (a machine learning tool).

**Usage**

```
decimal_scaling(x)
```

**Arguments**

x                   Attribute value(s).

**Value**

A decimal scaled value when applied to a single value or a decimal scaled attribute(s).

**Note**

The lapply function is required to apply this function across several attributes. This is NOT a normalization function, so data obtained from the decimal scaling function exists on a sliding scale and SHOULD NOT be used for meaningful analysis.

**Author(s)**

Tingwei Adeck

**References**

<https://www.statology.org/how-to-normalize-data-in-r/>

## Examples

```
test_df <- as.data.frame(c(seq(40)))
colnames(test_df) <- "test"
test_df_norm <- lapply(test_df[1:ncol(test_df)], decimal_scaling)
```

---

fluor\_threshold\_check *Title: A fluorescence quantification Quality Control (QC) function.*

---

## Description

A function designed to check that fluorescence values do not exceed the upper limit ( $2^{15}$  or 32768) OR fall below the lower limit ( $2^{11}$  or 2048). Fluorescence values that exceed these thresholds are considered noisy and lead to incorrect interpretation of analysis results.

## Usage

```
fluor_threshold_check(clean_df, fun = NA)
```

## Arguments

clean_df	A cleaned data frame.
fun	A parameter used for Boolean expressions.

## Value

A polite warning message to the data analyst or researcher.

## Note

Experimental issues should be investigated at very high or very low fluorescence values. The most common experimental issues arise when ACMA concentrations are out of the tolerated range. Based on my experience, ACMA concentrations between 2 and 5 Micromolar will suffice to get fluorescence values within the tolerance threshold. ACMA concentrations as low as 0.2 Micromolar or as high as 20 Micromolar have proven problematic based on my research experience.

A second issue linked to the FLUOstar instrument revolves around setting the right “gain” to ensure the right level of sensitivity in machine readings. A very high “gain” setting results in increased machine sensitivity even at the right ACMA concentrations and vice versa. In short, we want the machine to be primed to read exactly what we feed it, no more, no less.

This function provides the attribute(s) and tuple(s) for the values that need investigation.

These deductions were obtained from my experimental hiccups and my characterization of the liposome flux assay system.

## Author(s)

Tingwei Adeck



**See Also**

[fluor\\_threshold\\_check\\_na\(\)](#), [fluor\\_threshold\\_check\\_raw\(\)](#)

**Examples**

```
fpath <- system.file("extdata", "dat_1.dat", package = "normfluodbf", mustWork = TRUE)
dat_df <- read.table(file=fpath)
nocomma_dat <- clean_odddat_optimus(dat_df)
resampled_scaled <- resample_dat_scale(nocomma_dat, tnp=3, cycles=40)
resampled_scaled <- resampled_scaled[,c(1:4)]
fluor_threshold_check(resampled_scaled)
```

---

fluor\_threshold\_check\_na

*Title: A missing value (NA) Quality Control (QC) function.*

---

**Description**

A function designed to check for missing values in a data frame.

**Usage**

```
fluor_threshold_check_na(clean_df, fun = NA)
```

**Arguments**

clean_df	A cleaned data frame.
fun	A parameter used for Boolean expressions.

**Value**

A polite warning message to the data analyst or researcher.

**Author(s)**

Tingwei Adeck

**See Also**

[fluor\\_threshold\\_check\(\)](#), [fluor\\_threshold\\_check\\_raw\(\)](#)

**Examples**

```
fpath <- system.file("extdata", "dat_1.dat", package = "normfluodbf", mustWork = TRUE)
dat_df <- read.table(file=fpath)
nocomma_dat <- clean_odddat_optimus(dat_df)
fluor_threshold_check_na(nocomma_dat)
```

---

fluor\_threshold\_check\_raw

*Title: A fluorescence quantification Quality Control (QC) function.*

---

### Description

A function designed to check that fluorescence values do not exceed the upper limit ( $2^{15}$  or 32768) OR fall below the lower limit ( $2^{11}$  or 2048). Fluorescence values that exceed these thresholds are considered noisy and lead to incorrect interpretation of analysis results.

### Usage

```
fluor_threshold_check_raw(clean_df, fun = NA)
```

### Arguments

clean_df	A cleaned data frame.
fun	A parameter used for Boolean expressions.

### Value

A polite warning message to the data analyst or researcher.

### Note

This function works on a different type of data frame but in a similar manner to @seealso [fluor\\_threshold\\_check\(\)](#).

### Author(s)

Tingwei Adeck

### See Also

[fluor\\_threshold\\_check\(\)](#), [fluor\\_threshold\\_check\\_na\(\)](#)

### Examples

```
fpath <- system.file("extdata", "dat_1.dat", package = "normfluodbf", mustWork = TRUE)
dat_df <- read.table(file=fpath)
nocomma_dat <- clean_odddat_optimus(dat_df)
fluor_threshold_check_raw(nocomma_dat)
```

---

generic_identifier	<i>Title: A function to append a generic identifier attribute to any data frame, but users supply a name for said attribute.</i>
--------------------	--

---

**Description**

A function that creates an attribute of `seq(numrows)` with a step size of 1, where the user provides the attribute name.

**Usage**

```
generic_identifier(numrows, col_name)
```

**Arguments**

numrows	The number of rows the user intends to have in the created data frame.
col_name	The desired attribute name.

**Value**

A user-named single attribute data frame with `nrow = numrows`.

**Author(s)**

Tingwei Adeck

**Examples**

```
generic_identifier(40, col_name="Cycle_No")
```

---

ggplot_tnp	<i>Title: A Visualization function.</i>
------------	---

---

**Description**

A visualization function using `ggplot2`.

**Usage**

```
ggplot_tnp(df, x, y_list, xlim, ylim, xlab = NULL, ylab = NULL, title = NULL)
```

**Arguments**

<code>df</code>	A clean data frame with attributes or tuples containing a mixture of samples.
<code>x</code>	The X-variable often the cycle number OR time.
<code>y_list</code>	A character vector of samples that need to be plotted. Often of the format TNP (Test, Negative, Positive).
<code>xlim</code>	The X-variable scale.
<code>ylim</code>	The Y-variable scale.
<code>xlab</code>	The X axis label. Takes a string.
<code>ylab</code>	The Y-axis label. Takes a string.
<code>title</code>	Graph title. Takes a string.

**Value**

A nice visual of the clean and normalized data frame.

**Author(s)**

Tingwei Adeck

**Examples**

```
fpath <- system.file("extdata", "dat_1.dat", package = "normfluodbf", mustWork = TRUE)
dat_df <- normfluodat(fpath,3,40, rows_used = c('A','B','C'), norm_scale = 'one')
yvars <- c("A1","B1","C1")
xvar <- c("Cycle_Number", "Time")
xl <- c(0,40)
yl <- c(0,1)
ggplot_tnp(dat_df,xvar,yvars,xl,yl,xlab='Cycle_Number',ylab='Normalized Fluorescence',title='LFA')
```

---

liposomes\_214

*liposomes\_214.*

---

**Description**

FLUOstar .dbf file in wide format and unable to use for data analysis.

**Usage**

```
liposomes_214
```

**Format**

An object of class `data.frame` with 11 rows and 52 columns.

---

liposomes_215	<i>liposomes_215.</i>
---------------	-----------------------

---

**Description**

FLUOstar .dbf file in wide format and unable to use for data analysis.

**Usage**

liposomes\_215

**Format**

An object of class data.frame with 11 rows and 52 columns.

---

liposomes_216	<i>liposomes_216.</i>
---------------	-----------------------

---

**Description**

FLUOstar .dbf file in wide format and unable to use for data analysis.

**Usage**

liposomes\_216

**Format**

An object of class data.frame with 8 rows and 52 columns.

---

liposomes_218	<i>liposomes_218.</i>
---------------	-----------------------

---

**Description**

FLUOstar .dbf file in wide format and unable to use for data analysis.

**Usage**

liposomes\_218

**Format**

An object of class data.frame with 11 rows and 52 columns.

---

liposomes_221	<i>liposomes_221.</i>
---------------	-----------------------

---

**Description**

FLUOstar .dbf file in wide format and unable to use for data analysis.

**Usage**

liposomes\_221

**Format**

An object of class data.frame with 38 rows and 52 columns.

---

liposomes_227	<i>liposomes_227.</i>
---------------	-----------------------

---

**Description**

FLUOstar .dbf file in wide format and unable to use for data analysis.

**Usage**

liposomes\_227

**Format**

An object of class data.frame with 29 rows and 52 columns.

---

log_transformation	<i>Title: A log transformation function.</i>
--------------------	--

---

**Description**

Title: A log transformation function.

**Usage**

log\_transformation(x)

**Arguments**

x                   Attribute value(s).

**Value**

A log-transformed value when applied to a single value or an attribute with log-transformed values.

**Note**

The lapply function is required to apply this function across several attributes.

**Author(s)**

Tingwei Adeck

**References**

<https://www.statology.org/how-to-normalize-data-in-r/>

**Examples**

```
test_df <- as.data.frame(c(seq(40)))
colnames(test_df) <- "test"
test_df_norm <- lapply(test_df[1:ncol(test_df)], log_transformation)
```

---

min\_max\_norm

*Title: Min-Max normalization on a 0-1 scale.*

---

**Description**

Title: Min-Max normalization on a 0-1 scale.

**Usage**

```
min_max_norm(x)
```

**Arguments**

x                   Attribute value(s).

**Value**

A normalized value (between 0 and 1) when applied to a single value or a normalized attribute with values between 0 and 1.

**Note**

The lapply function is required to apply this function across several attributes.

**Author(s)**

Tingwei Adeck

**References**

<https://www.statology.org/how-to-normalize-data-in-r/>

**Examples**

```
test_df <- as.data.frame(c(seq(40)))
colnames(test_df) <- "test"
test_df_norm <- lapply(test_df[1:ncol(test_df)], min_max_norm)
```

---

min\_max\_norm\_percent *Title: Min-Max normalization on a 0-100 scale.*

---

**Description**

Title: Min-Max normalization on a 0-100 scale.

**Usage**

```
min_max_norm_percent(x)
```

**Arguments**

x                   Attribute value(s).

**Value**

A normalized value (between 0 and 100) when applied to a single value or a normalized attribute with values between 0 and 100.

**Note**

The lapply function is required to apply this function across several attributes.

**Author(s)**

Tingwei Adeck

**References**

<https://www.statology.org/how-to-normalize-data-in-r/>

**Examples**

```
test_df <- as.data.frame(c(seq(40)))
colnames(test_df) <- "test"
test_df_norm <- lapply(test_df[1:ncol(test_df)], min_max_norm_percent)
```



normfluodat

*Title: Cleans and normalizes DAT files obtained from experiments using the FLUOstar Omega microplate reader (from BMG LABTECH).*

## Description

The simplest case scenario entails inputting the name or directory of a DAT file as a string, the number of rows denoted by the tnp (test, negative, positive) parameter, and the number of cycles (selected by the user when running the FLUOstar instrument). The program takes these three base-line parameters, performs cleaning and normalization of the DAT file, and then appends an attribute called "Cycle\_Number" to the normalized data frame.

## Usage

```
normfluodat(
  dat,
  tnp,
  cycles,
  rows_used = NULL,
  cols_used = NULL,
  user_specific_labels = NULL,
  read_direction = NULL,
  norm_scale = NULL,
  interval = NULL,
  first_end = NULL,
  pause_duration = NULL,
  end_time = NULL
)
```

## Arguments

dat	A string ("dat_1.dat") if the file is found within the present working directory (pwd) OR a path pointing directly to a ".dat" file.
tnp	A numeric value indicating the number of rows used. TNP is used as an acronym for Test, Negative, Positive.
cycles	A numeric value indicating the number of cycles selected by the user when running the FLUOstar instrument.
rows_used	A character vector of the rows used; ru = c('A','B','C').
cols_used	A numeric vector of the columns used; cu = c(1,2,3).
user_specific_labels	A character vector manually prepared by the user to denote the wells used on the microplate reader; usl = c('A1','B1','C1').
read_direction	A string input with two choices, "vertical" or "horizontal." The user indicates "vertical" if the user intends to have a final data frame with samples arranged as sample type triplets (A1, B1, C1, A1, B1, C1) OR "horizontal" if the user

	intends to have a final data frame with samples arranged as clusters per sample type (A1, A2, A3, B1, B2, B3).
norm_scale	This parameter takes sub-parameters: 'raw' , 'hundred' , 'one' , 'z-score' , or 'decimal' , which denotes the normalization type or scale; Initialized as NULL.
interval	The time interval chosen for the assay often in seconds.
first_end	The end time of the initial run, often the pause for the introduction of a new substance. This can be the cycle number chosen for the initial stop.
pause_duration	The time between the first end (pause) and resumption of the assay.
end_time	The final end time of the assay.

**Value**

A normalized data frame with an appended "Cycle\_Number" attribute. The "Cycle\_Number" attribute is the X-variable.

**Note**

This function is a single-step function leveraging several subordinate functions. It is assumed that the user has the 3 baseline parameters to get this function working. Users must double-check attribute names to ensure they end up with accurate results.

**Author(s)**

Tingwei Adeck

**See Also**

[normfluodatlite\(\)](#)

**Examples**

```
fpath <- system.file("extdata", "dat_4.dat", package = "normfluodbf", mustWork = TRUE)
normalized_fluo_dat <- normfluodat(dat=fpath, tnp = 3, cycles = 40)
```

---

normfluodatfull	<i>Title: Cleans and normalizes DAT files obtained from experiments using the FLUOstar Omega microplate reader (from BMG LABTECH).</i>
-----------------	--

---

**Description**

The simplest case scenario entails inputting the name or directory of a DAT file as a string, the number of rows denoted by the tnp (test, negative, positive) parameter, and the number of cycles (selected by the user when running the FLUOstar instrument). The program takes these three baseline parameters, performs cleaning and normalization of the DAT file, and then appends an attribute called "Cycle\_Number" to the normalized data frame.

**Usage**

```
normfluodatfull(
  dat,
  tnp,
  cycles,
  rows_used = NULL,
  cols_used = NULL,
  user_specific_labels = NULL,
  read_direction = NULL,
  norm_scale = NULL,
  na_omit = NULL
)
```

**Arguments**

dat	A string ("dat_1.dat") if the file is found within the present working directory (pwd) OR a path pointing directly to a ".dat" file.
tnp	A numeric value indicating the number of rows used. TNP is used as an acronym for Test, Negative, Positive.
cycles	A numeric value indicating the number of cycles selected by the user when running the FLUOstar instrument.
rows_used	A character vector of the rows used; ru = c('A','B','C').
cols_used	A numeric vector of the columns used; cu = c(1,2,3).
user_specific_labels	A character vector manually prepared by the user to denote the wells used on the microplate reader; usl = c('A1','B1','C1').
read_direction	A string input with two choices, "vertical" or "horizontal." The user indicates "vertical" if the user intends to have a final data frame with samples arranged as sample type triplets (A1, B1, C1, A1, B1, C1) OR "horizontal" if the user intends to have a final data frame with samples arranged as clusters per sample type (A1, A2, A3, B1, B2, B3).
norm_scale	This parameter takes sub-parameters: 'raw' , 'hundred' , 'one' , 'z-score' , or 'decimal' , which denotes the normalization type or scale; Initialized as NULL.
na_omit	Takes a string "yes" OR "no".

**Value**

A normalized data frame with an appended "Cycle\_Number" attribute. The "Cycle\_Number" attribute is the X-variable.

**Note**

This function is a single-step function leveraging several subordinate functions. It is assumed that the user has the 3 baseline parameters to get this function working. Users must double-check attribute names to ensure they end up with accurate results.

**Author(s)**

Tingwei Adeck

**See Also**[normfluodat\(\)](#)**Examples**

```
fpath <- system.file("extdata", "dat_1.dat", package = "normfluodbf", mustWork = TRUE)
normalized_fluo_dat <- normfluodatfull(dat=fpath, tnp = 3, cycles = 40)
```

normfluodatlite

---

*Title: Cleans and normalizes DAT files obtained from experiments using the FLUOstar Omega microplate reader (from BMG LABTECH).*

---

**Description**

The simplest case scenario entails inputting the name or directory of a DAT file as a string, the number of rows denoted by the tnp (test, negative, positive) parameter, and the number of cycles (selected by the user when running the FLUOstar instrument). The program takes these three baseline parameters, performs cleaning and normalization of the DAT file, and then appends an attribute called "Cycle\_Number" to the normalized data frame.

**Usage**

```
normfluodatlite(
  dat,
  tnp,
  cycles,
  rows_used = NULL,
  cols_used = NULL,
  user_specific_labels = NULL,
  read_direction = NULL,
  norm_scale = NULL
)
```

**Arguments**

dat	A string ("dat_1.dat") if the file is found within the present working directory (pwd) OR a path pointing directly to a ".dat" file.
tnp	A numeric value indicating the number of rows used. TNP is used as an acronym for Test, Negative, Positive.
cycles	A numeric value indicating the number of cycles selected by the user when running the FLUOstar instrument.
rows_used	A character vector of the rows used; ru = c('A','B','C').

cols_used	A numeric vector of the columns used; cu = c(1,2,3).
user_specific_labels	A character vector manually prepared by the user to denote the wells used on the microplate reader; usl = c('A1','B1','C1').
read_direction	A string input with two choices, "vertical" or "horizontal." The user indicates "vertical" if the user intends to have a final data frame with samples arranged as sample type triplets (A1, B1, C1, A1, B1, C1) OR "horizontal" if the user intends to have a final data frame with samples arranged as clusters per sample type (A1, A2, A3, B1, B2, B3).
norm_scale	This parameter takes sub-parameters: 'raw' , 'hundred' , 'one' , 'z-score' , or 'decimal' , which denotes the normalization type or scale; Initialized as NULL.

**Value**

A normalized data frame with an appended "Cycle\_Number" attribute. The "Cycle\_Number" attribute is the X-variable.

**Note**

This function is a single-step function leveraging several subordinate functions. It is assumed that the user has the 3 baseline parameters to get this function working. Users must double-check attribute names to ensure they end up with accurate results.

**Author(s)**

Tingwei Adeck

**See Also**

[normfluodat\(\)](#)

**Examples**

```
fpath <- system.file("extdata", "dat_1.dat", package = "normfluodbf", mustWork = TRUE)
normalized_fluo_dat <- normfluodatlite(dat=fpath, tnp = 3, cycles = 40)
```

---

normfluordat

*Title: Cleans and normalizes DAT files obtained from experiments using the FLUOstar Omega microplate reader (from BMG LABTECH).*

---

**Description**

The simplest case scenario entails inputting the name or directory of a DAT file as a string, the number of rows denoted by the tnp (test, negative, positive) parameter, and the number of cycles (selected by the user when running the FLUOstar instrument). The program takes these three baseline parameters, performs cleaning and normalization of the DAT file, and then appends an attribute called "Cycle\_Number" to the normalized data frame.

**Usage**

```
normfluordat(
  dat,
  tnp,
  cycles,
  rows_used = NULL,
  cols_used = NULL,
  user_specific_labels = NULL,
  read_direction = NULL,
  na_omit = NULL
)
```

**Arguments**

dat	A string ("dat_1.dat") if the file is found within the present working directory (pwd) OR a path pointing directly to a ".dat" file.
tnp	A numeric value indicating the number of rows used. TNP is used as an acronym for Test, Negative, Positive.
cycles	A numeric value indicating the number of cycles selected by the user when running the FLUOstar instrument.
rows_used	A character vector of the rows used; ru = c('A','B','C').
cols_used	A numeric vector of the columns used; cu = c(1,2,3).
user_specific_labels	A character vector manually prepared by the user to denote the wells used on the microplate reader; usl = c('A1','B1','C1').
read_direction	A string input with two choices, "vertical" or "horizontal." The user indicates "vertical" if the user intends to have a final data frame with samples arranged as sample type triplets (A1, B1, C1, A1, B1, C1) OR "horizontal" if the user intends to have a final data frame with samples.
na_omit	Takes a string "yes" OR "no".

**Value**

A normalized data frame with an appended "Cycle\_Number" attribute. The "Cycle\_Number" attribute is the X-variable.

**Note**

This function has less optimized space and time complexities than @seealso [normfluodat\(\)](#). In real-time the difference in optimization is not noticeable. This function also takes less parameters than the more optimized version of the function. Use @seealso [normfluodat\(\)](#) for better approximation of attribute names.

**Author(s)**

Tingwei Adeck

**See Also**[normfluodat\(\)](#)**Examples**

```
fpath <- system.file("extdata", "dat_1.dat", package = "normfluordbf", mustWork = TRUE)
normalized_fluo_dat <- normfluodat(dat=fpath, tnp = 3, cycles = 40)
```

normfluordbf

*Title: Cleans and Normalizes DBF files obtained from experiments using the FLUOstar microplate reader.*

**Description**

Input the path to a ".dbf" file obtained from the FLUOstar microplate (usually a 96-well microplate) reader; this function will create a data frame, clean the data frame, normalize the data frame, append a "Cycle\_Number" column and return a data frame that is ready for analysis. Most importantly, this function is a single\_step function. Also, the function can be extended to other ".dbf" files if they follow the format for which this function was designed; this is totally at the users' discretion.

**Usage**

```
normfluordbf(file = NULL, norm_scale = NULL, transformed = NULL, fun = NA, ...)
```

**Arguments**

file	A string ("liposomes_xxx.dbf") if the file is found within the present working directory (pwd) OR a path pointing directly to a ".dbf" file, from FLUOstar experiments.
norm_scale	This parameter can taken in 'hundred', 'one', or 'z-score' which denotes the normalization type; Initialized as NULL.
transformed	This parameter can take in 'log' which denotes a logarithmic box-cox transformation; Initialized as NULL.
fun	A variable defined as NA, used for boolean expressions or manipulation.
...	A container object that can be used to capture extra variables if needed.

**Value**

A normalized data frame with an appended "Cycle\_Number" attribute.

**Note**

Re-nomenclature of norm\_tidy\_dbf to a more appropriate name that facilitates function utilization. Users can continue with the old name ("norm\_tidy\_dbf") but this is a better name in my opinion.

**Author(s)**

Tingwei Adeck

**Examples**

```
fpath <- system.file("extdata", "liposomes_214.dbf", package = "normfluordbf", mustWork = TRUE)
normalized_dbf <- normfluordbf(file=fpath, norm_scale = 'raw')
```

---

norm\_applier

*Title: A normalization applier built on lapply.*

---

**Description**

Applies a function over a list of attributes.

**Usage**

```
norm_applier(df, norm_scale = c("one", "hundred", "z-score", "raw", "decimal"))
```

**Arguments**

df                    A data frame.

norm\_scale           This parameter takes sub-parameters: 'raw', 'hundred', 'one', 'z-score', or 'decimal', which denotes the normalization type or scale.

**Value**

A data frame with attribute values obtained from the applied function using lapply.

**Examples**

```
test_df <- as.data.frame(c(seq(40)))
colnames(test_df) <- "test"
test_df_norm <- norm_applier(test_df, norm_scale = 'one')
```



norm\_tidy\_dbf

*Title: Cleans and Normalizes DBF files obtained from experiments using the FLUOstar Omega microplate reader (from BMG LABTECH).*

## Description

The simplest function utilization scenario entails an input of the path to a DBF file obtained from the FLUOstar microplate (usually a 96-well microplate) reader; In a single step, this function will create a data frame, clean the data frame, normalize the data frame, append a "Cycle\_Number" attribute, perform an adjustment to the "time" attribute and return a data frame that is ready for analysis. Since the initial publication of this package, several changes have been made to improve the user experience and to give the user more options to fine-tune the output from the package to meet the users' aesthetic needs. Users who decide to move past the simplest utility scenario have been given more options to customize the output based on the users' needs. Notably, several normalization sub-parameters have been provided in the package which yields different outputs based on what the user is used to seeing. Just as the FLUOstar instrument is built to handle an array of assays, this function is designed to be multi-dimensional (meaning it can handle data with the same DBF extension from other assay types), on the condition that the data from assay types other than liposome flux assays follow the same data format this package was designed to handle. Of course, users of this package are advised to pre-analyze DBF files from other assay types to ensure they are compliant with this package (compliance in this scenario is simple meaning DBF files from other assays should be like DBF files from liposome flux assays).

## Usage

```
norm_tidy_dbf(
  file = NULL,
  norm_scale = NULL,
  transformed = NULL,
  fun = NA,
  ...
)
```

## Arguments

file	A string ("liposomes_xxx.dbf") if the file is found within the present working directory (pwd) OR a path pointing directly to a ".dbf" file.
norm_scale	This parameter takes sub-parameters: 'raw' , 'hundred' , 'one' , 'z-score' , or 'decimal' , which denotes the normalization type or scale; The parameter is initialized as NULL.
transformed	This parameter takes input 'log' , which denotes a logarithmic box-cox transformation; Initialized as NULL.
fun	A parameter defined as NA is used for Boolean expressions or manipulation.
...	An abstract placeholder or container parameter that can be used to capture extra variables if needed.

**Value**

A normalized data frame with an appended "Cycle\_Number" attribute.

**Note**

The default normalization sub-parameter outputs values in the 0-1 range. Unless a "norm\_scale" level is specified by the user, the default output is in the 0-1 range. The "norm\_scale" sub-parameter "decimal" is a machine-learning tool and should be avoided; it also provides no advantage for basic research analysis as its output operates on a sliding scale just like the raw data. Logarithmic transformation provides a minuscule advantage in data analysis and could/should be avoided. Backward compatibility is maintained in all updates, so there should be no issues with using the package the way the user was used to. The favorite "norm\_scale" level is "z-score" since it divides the axis into negative and positive, thus facilitating interpretation.

**Author(s)**

Tingwei Adeck

**See Also**

[normfluordbf\(\)](#), [normfluodat\(\)](#)

**Examples**

```
fpath <- system.file("extdata", "liposomes_214.dbf", package = "normfluordbf", mustWork = TRUE)
normalized_dbf <- norm_tidy_dbf(file=fpath, norm_scale = 'raw')
```

---

norm\_z

*Title: Z-score standardization or normalization function.*

---

**Description**

Title: Z-score standardization or normalization function.

**Usage**

```
norm_z(x)
```

**Arguments**

x                   Attribute value(s).

**Value**

A standardized value ( $Z = N(0,1)$ ) when applied to a single value or a standardized attribute with values ( $Z = N(0,1)$ ).

**Note**

The lapply function is required to apply this function across several attributes.

**Author(s)**

Tingwei Adeck

**References**

<https://www.statology.org/how-to-normalize-data-in-r/>

**Examples**

```
test_df <- as.data.frame(c(seq(40)))
colnames(test_df) <- "test"
test_df_norm <- lapply(test_df[1:ncol(test_df)], norm_z)
```

---

resample\_dat

*Title: A function to create an attribute or column for each sample loaded into the microplate wells.*

---

**Description**

Designed as a prototype function to take a single attribute or column consisting of several samples and perform a putative resampling to yield another data frame with a separate attribute for each sample.

**Usage**

```
resample_dat(df, tnp, cycles)
```

**Arguments**

df	A clean data frame with attributes or tuples containing a mixture of samples.
tnp	A numeric value indicating the number of rows used. TNP is used as an acronym for Test, Negative, Positive.
cycles	A numeric value indicating the number of cycles selected by the user when running the FLUOstar instrument.

**Value**

A new data frame where separated samples are assigned a separate attribute or column.

**Author(s)**

Tingwei Adeck

**See Also**

[resample\\_dat\\_scale\(\)](#), [resample\\_dat\\_scale\\_optimus\(\)](#)

**Examples**

```
fpath <- system.file("extdata", "dat_5.dat", package = "normfluodbf", mustWork = TRUE)
dat_df <- read.table(file=fpath)
nocomma_dat <- clean_odddat_optimus(dat_df)
samples_delineated <- resample_dat(nocomma_dat, tnp=3, cycles=40)
```

---

resample_dat_alt	<i>Title: A function to create an attribute or column for each sample loaded into the microplate wells.</i>
------------------	---

---

**Description**

: Designed as a prototype function to take a single tuple or row consisting of several samples and perform a putative resampling to yield another data frame with a separate attribute for each sample.

**Usage**

```
resample_dat_alt(df, tnp, cycles)
```

**Arguments**

df	A clean data frame with attributes or tuples containing a mixture of samples.
tnp	A numeric value indicating the number of rows used. TNP is used as an acronym for Test, Negative, Positive.
cycles	A numeric value indicating the number of cycles selected by the user when running the FLUOstar instrument.

**Value**

A new data frame where separated samples are assigned a separate attribute or column.

**Author(s)**

Tingwei Adeck

**See Also**

[resample\\_dat\\_scale\\_alt\(\)](#)

**Examples**

```
fpath <- system.file("extdata", "dat_5.dat", package = "normfluodbf", mustWork = TRUE)
dat_df <- read.table(file=fpath)
nocomma_dat <- clean_odddat_optimus(dat_df)
samples_delineated <- resample_dat_alt(nocomma_dat, tnp=3, cycles=40)
```

---

resample_dat_scale	<i>Title: A function to create an attribute or column for each sample loaded into the microplate wells.</i>
--------------------	---

---

### Description

Creates a data frame where each sample loaded into the microplate wells has a separate attribute.

### Usage

```
resample_dat_scale(df, tnp, cycles)
```

### Arguments

df	A clean data frame with attributes or tuples containing a mixture of samples.
tnp	A numeric value indicating the number of rows used. TNP is used as an acronym for Test, Negative, Positive.
cycles	A numeric value indicating the number of cycles selected by the user when running the FLUOstar instrument.

### Value

A new data frame where separated samples are assigned a separate attribute or column.

### Note

This function builds on or scales-up @seealso [resample\\_dat\(\)](#), hence the suffix scale. This function is less optimized than @seealso [resample\\_dat\\_scale\\_optimus\(\)](#).

### Author(s)

Tingwei Adeck

### See Also

[resample\\_dat\(\)](#)

### Examples

```
fpath <- system.file("extdata", "dat_4.dat", package = "normfluodbf", mustWork = TRUE)
dat_df <- read.table(file=fpath)
nocomma_dat <- clean_odd_dat_optimus(dat_df)
resampled_scaled <- resample_dat_scale(nocomma_dat, tnp=3, cycles=40)
```

---

`resample_dat_scale_alt`

*Title: A function to create an attribute or column for each sample loaded into the microplate wells.*

---

### Description

A function that takes tuples or rows consisting of several samples and perform a putative resampling to yield another data frame with a separate attribute for each sample.

### Usage

```
resample_dat_scale_alt(df, tnp, cycles, na_omit = NULL)
```

### Arguments

<code>df</code>	A clean data frame with attributes or tuples containing a mixture of samples.
<code>tnp</code>	A numeric value indicating the number of rows used. TNP is used as an acronym for Test, Negative, Positive.
<code>cycles</code>	A numeric value indicating the number of cycles selected by the user when running the FLUOstar instrument.
<code>na_omit</code>	Takes a string "yes" OR "no".

### Value

A new data frame where separated samples are assigned a separate attribute or column.

### Author(s)

Tingwei Adeck

### See Also

[resample\\_dat\\_alt\(\)](#)

### Examples

```
fpath <- system.file("extdata", "dat_1.dat", package = "normfluodbf", mustWork = TRUE)
dat_df <- read.table(file=fpath)
nocomma_dat <- clean_odddat_optimus(dat_df)
resampled_scaled <- resample_dat_scale_alt(nocomma_dat, tnp=3, cycles=40)
```

---

`resample_dat_scale_alt_bfv`

*Title: A function to create an attribute or column for each sample loaded into the microplate wells.*

---

## Description

A function that takes tuples or rows consisting of several samples and perform a putative resampling to yield another data frame with a separate attribute for each sample.

## Usage

```
resample_dat_scale_alt_bfv(df, tnp, cycles)
```

## Arguments

<code>df</code>	A clean data frame with attributes or tuples containing a mixture of samples.
<code>tnp</code>	A numeric value indicating the number of rows used. TNP is used as an acronym for Test, Negative, Positive.
<code>cycles</code>	A numeric value indicating the number of cycles selected by the user when running the FLUOstar instrument.

## Value

A new data frame where separated samples are assigned a separate attribute or column.

## Author(s)

Tingwei Adeck

## See Also

[resample\\_dat\\_alt\(\)](#), [resample\\_dat\\_scale\\_alt\(\)](#)

## Examples

```
fpath <- system.file("extdata", "dat_4.dat", package = "normfluodbf", mustWork = TRUE)
dat_df <- read.table(file=fpath)
nocomma_dat <- clean_odddat_optimus(dat_df)
resampled_scaled <- resample_dat_scale_alt_bfv(nocomma_dat, tnp=3, cycles=40)
```

---

resample\_dat\_scale\_alt\_bf\_na

*Title: A function to create an attribute or column for each sample loaded into the microplate wells.*

---

## Description

A function that takes tuples or rows consisting of several samples and perform a putative resampling to yield another data frame with a separate attribute for each sample.

## Usage

```
resample_dat_scale_alt_bf_na(df, tnp, cycles)
```

## Arguments

df	A clean data frame with attributes or tuples containing a mixture of samples.
tnp	A numeric value indicating the number of rows used. TNP is used as an acronym for Test, Negative, Positive.
cycles	A numeric value indicating the number of cycles selected by the user when running the FLUOstar instrument.

## Value

A new data frame where separated samples are assigned a separate attribute or column.

## Author(s)

Tingwei Adeck

## See Also

[resample\\_dat\\_alt\(\)](#), [resample\\_dat\\_scale\\_alt\(\)](#)

## Examples

```
fpath <- system.file("extdata", "dat_4.dat", package = "normfluodbf", mustWork = TRUE)
dat_df <- read.table(file=fpath)
nocomma_dat <- clean_odddat_optimus(dat_df)
resampled_scaled <- resample_dat_scale_alt_bf_na(nocomma_dat, tnp=3, cycles=40)
```



---

`resample_dat_scale_alt_na`

*Title: A function to create an attribute or column for each sample loaded into the microplate wells.*

---

### Description

A function that takes tuples or rows consisting of several samples and perform a putative resampling to yield another data frame with a separate attribute for each sample. NA values are retained.

### Usage

```
resample_dat_scale_alt_na(df, tnp, cycles)
```

### Arguments

<code>df</code>	A clean data frame with attributes or tuples containing a mixture of samples.
<code>tnp</code>	A numeric value indicating the number of rows used. TNP is used as an acronym for Test, Negative, Positive.
<code>cycles</code>	A numeric value indicating the number of cycles selected by the user when running the FLUOstar instrument.

### Value

A new data frame where separated samples are assigned a separate attribute or column.

### Author(s)

Tingwei Adeck

### See Also

[resample\\_dat\\_alt\(\)](#)

### Examples

```
fpath <- system.file("extdata", "dat_1.dat", package = "normfluodbf", mustWork = TRUE)
dat_df <- read.table(file=fpath)
nocomma_dat <- clean_odddat_optimus(dat_df)
resampled_scaled <- resample_dat_scale_alt_na(nocomma_dat, tnp=3, cycles=40)
```

---

resample\_dat\_scale\_naretainer

*Title: A function to create an attribute or column for each sample loaded into the microplate wells.*

---

## Description

Creates a data frame where each sample loaded into the microplate wells has a separate attribute. NA values are retained for more control.

## Usage

```
resample_dat_scale_naretainer(df, tnp, cycles)
```

## Arguments

**df** A clean data frame with attributes or tuples containing a mixture of samples.

**tnp** A numeric value indicating the number of rows used. TNP is used as an acronym for Test, Negative, Positive.

**cycles** A numeric value indicating the number of cycles selected by the user when running the FLUOstar instrument.

## Value

A new data frame where separated samples are assigned a separate attribute or column.

## Note

This function builds on or scales-up @seealso [resample\\_dat\(\)](#), hence the suffix scale. This function is less optimized than @seealso [resample\\_dat\\_scale\\_optimus\(\)](#).

## Author(s)

Tingwei Adeck

## See Also

[resample\\_dat\(\)](#)

## Examples

```
fpath <- system.file("extdata", "dat_4.dat", package = "normfluodbf", mustWork = TRUE)
dat_df <- read.table(file=fpath)
nocomma_dat <- clean_odddat_optimus(dat_df)
resampled_scaled <- resample_dat_scale_naretainer(nocomma_dat, tnp=3, cycles=40)
```

---

`resample_dat_scale_optimus`

*Title: A function to create an attribute or column for each sample loaded into the microplate wells.*

---

**Description**

Creates a data frame where each sample loaded into the microplate wells has a separate attribute.

**Usage**

```
resample_dat_scale_optimus(df, tnp, cycles)
```

**Arguments**

<code>df</code>	A clean data frame with attributes or tuples containing a mixture of samples.
<code>tnp</code>	A numeric value indicating the number of rows used. TNP is used as an acronym for Test, Negative, Positive.
<code>cycles</code>	A numeric value indicating the number of cycles selected by the user when running the FLUOstar instrument.

**Value**

A new data frame where separated samples are assigned a separate attribute or column.

**Note**

This function builds on or scales-up @seealso [resample\\_dat\(\)](#), hence the suffix scale. This function is more optimized than @seealso [resample\\_dat\\_scale\(\)](#), hence the suffix scale\_optimus.

**Author(s)**

Tingwei Adeck

**See Also**

[resample\\_dat\(\)](#)

**Examples**

```
fpath <- system.file("extdata", "dat_1.dat", package = "normfluodbf", mustWork = TRUE)
dat_df <- read.table(file=fpath)
nocomma_dat <- clean_odd_dat_optimus(dat_df)
resampled_scaled <- resample_dat_scale_optimus(nocomma_dat, tnp=3, cycles=40)
```

---

`resample_dat_scale_optimus_backend`

*Title: A function to create an attribute or column for each sample loaded into the microplate wells.*

---

## Description

Creates a data frame where each sample loaded into the microplate wells has a separate attribute.

## Usage

```
resample_dat_scale_optimus_backend(df, tnp, cycles, na_omit = NULL)
```

## Arguments

<code>df</code>	A clean data frame with attributes or tuples containing a mixture of samples.
<code>tnp</code>	A numeric value indicating the number of rows used. TNP is used as an acronym for Test, Negative, Positive.
<code>cycles</code>	A numeric value indicating the number of cycles selected by the user when running the FLUOstar instrument.
<code>na_omit</code>	Takes a string "yes" OR "no".

## Value

A new data frame where separated samples are assigned a separate attribute or column.

## Note

This function builds on or scales-up @seealso [resample\\_dat\(\)](#), hence the suffix `scale`. This function is more optimized than @seealso [resample\\_dat\\_scale\(\)](#), hence the suffix `scale_optimus`.

## Author(s)

Tingwei Adeck

## See Also

[resample\\_dat\(\)](#)

## Examples

```
fpath <- system.file("extdata", "dat_1.dat", package = "normfluodbf", mustWork = TRUE)
dat_df <- read.table(file=fpath)
nocomma_dat <- clean_odddat_optimus(dat_df)
resampled_scaled <- resample_dat_scale_optimus_backend(nocomma_dat, tnp=3, cycles=40)
```

---

`resample_dat_scale_optimus_na`

*Title: A function to create an attribute or column for each sample loaded into the microplate wells.*

---

## Description

Creates a data frame where each sample loaded into the microplate wells has a separate attribute. NA values are retained.

## Usage

```
resample_dat_scale_optimus_na(df, tnp, cycles)
```

## Arguments

<code>df</code>	A clean data frame with attributes or tuples containing a mixture of samples.
<code>tnp</code>	A numeric value indicating the number of rows used. TNP is used as an acronym for Test, Negative, Positive.
<code>cycles</code>	A numeric value indicating the number of cycles selected by the user when running the FLUOstar instrument.

## Value

A new data frame where separated samples are assigned a separate attribute or column.

## Note

This function builds on or scales-up @seealso [resample\\_dat\(\)](#), hence the suffix `scale`. This function is more optimized than @seealso [resample\\_dat\\_scale\(\)](#), hence the suffix `scale_optimus`.

## Author(s)

Tingwei Adeck

## See Also

[resample\\_dat\(\)](#)

## Examples

```
fpath <- system.file("extdata", "dat_1.dat", package = "normfluodbf", mustWork = TRUE)
dat_df <- read.table(file=fpath)
nocomma_dat <- clean_odddat_optimus(dat_df)
resampled_scaled <- resample_dat_scale_optimus_na(nocomma_dat, tnp=3, cycles=40)
```

---

roundfluo*r*                      *Title: A value rounding function.*

---

**Description**

Round attribute values to three decimal places.

**Usage**

```
roundfluor(x)
```

**Arguments**

x                      Attribute value(s).

**Value**

A rounded value with three decimal places when applied to a single value or an attribute with log-transformed values.

**Examples**

```
test_df <- as.data.frame(c(seq(40)))  
colnames(test_df) <- "test"  
test_df_norm <- lapply(test_df[1:ncol(test_df)], roundfluor)
```

---

time\_attribute                      *Title: A time attribute generating function*

---

**Description**

A function to help the researcher reproduce the time attribute.

**Usage**

```
time_attribute(  
  interval = NULL,  
  first_end = NULL,  
  pause_duration = NULL,  
  end_time = NULL,  
  cycles = NULL  
)
```

**Arguments**

interval	The time interval chosen for the assay often in seconds.
first_end	The end time of the initial run, often the pause for the introduction of a new substance. This can be the cycle number chosen for the initial stop.
pause_duration	The time between the first end (pause) and resumption of the assay.
end_time	The final end time of the assay.
cycles	The number of cycles in the assay as selected by the user or researcher.

**Value**

The time attribute.

**Note**

The original function had an option for minutes which was for less time conscious people but the final version for this package has no such option. Users **MUST** provide numbers in seconds.

**Examples**

```
time_test = time_attribute(30,8,136,1276,40)
time_test = time_attribute(60,8,136,2460,40)
```

---

unique_identifier	<i>Title: A function to append a unique identifier attribute to any data frame within the normfluodbf package.</i>
-------------------	--

---

**Description**

The function in the context of normfluodbf creates an attribute called Cycle\_Number and appends this attribute to the cleaned or wrangled data frame derived from the dirty DBF file.

**Usage**

```
unique_identifier(df)
```

**Arguments**

df	A data frame with 1:n number of rows.
----	---------------------------------------

**Value**

A data frame with the Cycle\_Number attribute appended to the end of the data frame.

**Note**

The function operates in a closed system, meaning it is primarily designed to work with this package **ONLY**. Other use cases are simply a coincidence.

**Author(s)**

Tingwei Adeck

**See Also**

[normfluodat\(\)](#), [norm\\_tidy\\_dbf\(\)](#), [normfluordbf\(\)](#), [generic\\_identifier\(\)](#)

**Examples**

```
test_df <- as.data.frame(c(seq(40)))  
colnames(test_df) <- "test"  
unique_identifier(test_df)
```



# Index

## \* datasets

- dat\_1, [7](#)
- dat\_2, [8](#)
- dat\_3, [8](#)
- dat\_4, [8](#)
- dat\_5, [9](#)
- dat\_6, [9](#)
- dat\_7, [9](#)
- liposomes\_214, [20](#)
- liposomes\_215, [21](#)
- liposomes\_216, [21](#)
- liposomes\_218, [21](#)
- liposomes\_221, [22](#)
- liposomes\_227, [22](#)

actual\_cols\_used, [3](#)

check\_dat, [3](#)

check\_dbf, [4](#)

clean\_odd\_cc, [6](#)

clean\_odd\_cc(), [5](#), [7](#)

clean\_odddat\_optimus, [5](#)

clean\_odddat\_optimus(), [6](#), [7](#)

comma\_cleaner, [6](#)

comma\_cleaner(), [5](#), [6](#)

dat\_1, [7](#)

dat\_2, [8](#)

dat\_3, [8](#)

dat\_4, [8](#)

dat\_5, [9](#)

dat\_6, [9](#)

dat\_7, [9](#)

dat\_col\_names\_horizontal, [10](#)

dat\_col\_names\_optimus, [11](#)

dat\_col\_names\_optimus(), [15](#)

dat\_col\_names\_prime, [12](#)

dat\_col\_names\_rigid, [13](#)

dat\_col\_names\_rigid(), [12](#)

decimal\_scaling, [15](#)

fluor\_threshold\_check, [16](#)

fluor\_threshold\_check(), [17](#), [18](#)

fluor\_threshold\_check\_na, [17](#)

fluor\_threshold\_check\_na(), [17](#), [18](#)

fluor\_threshold\_check\_raw, [18](#)

fluor\_threshold\_check\_raw(), [17](#)

generic\_identifier, [19](#)

generic\_identifier(), [48](#)

ggplot\_tnp, [19](#)

liposomes\_214, [20](#)

liposomes\_215, [21](#)

liposomes\_216, [21](#)

liposomes\_218, [21](#)

liposomes\_221, [22](#)

liposomes\_227, [22](#)

log\_transformation, [22](#)

min\_max\_norm, [23](#)

min\_max\_norm\_percent, [24](#)

norm\_applier, [32](#)

norm\_tidy\_dbf, [33](#)

norm\_tidy\_dbf(), [48](#)

norm\_z, [34](#)

normfluodat, [25](#)

normfluodat(), [12](#), [28–31](#), [34](#), [48](#)

normfluodatfull, [26](#)

normfluodatlite, [28](#)

normfluodatlite(), [26](#)

normfluordat, [29](#)

normfluordbf, [31](#)

normfluordbf(), [34](#), [48](#)

resample\_dat, [35](#)

resample\_dat(), [37](#), [42–45](#)

resample\_dat\_alt, [36](#)

resample\_dat\_alt(), [38–41](#)

resample\_dat\_scale, [37](#)

resample\_dat\_scale(), [36](#), [43–45](#)

resample\_dat\_scale\_alt, 38  
resample\_dat\_scale\_alt(), 36, 39, 40  
resample\_dat\_scale\_alt\_bf\_na, 40  
resample\_dat\_scale\_alt\_bfv, 39  
resample\_dat\_scale\_alt\_na, 41  
resample\_dat\_scale\_naretain, 42  
resample\_dat\_scale\_optimus, 43  
resample\_dat\_scale\_optimus(), 36, 37, 42  
resample\_dat\_scale\_optimus\_backend, 44  
resample\_dat\_scale\_optimus\_na, 45  
roundflor, 46  
  
time\_attribute, 46  
  
unique\_identifer, 47